

IN THE CLAIMS

1 1. [currently amended] A method for securely communicating packets between a
2 first computer device and a second computer device through a packet-switched data
3 transmission network comprising intermediate computer devices, where at least one of
4 said computer devices performs a network address translation and/or a protocol
5 conversion, the method comprising the steps of

6 - determining what network address translations and/or protocol conversions, if
7 any, occur on packets transmitted between the first computer device and the
8 second computer device,

9 - if it is found that network address translations and/or protocol conversions
10 occur in a data path between said first computer device and a second computer
11 device, taking packets conforming to a first protocol and encapsulating them into
12 packets conforming to a second protocol, which second protocol is capable of
13 traversing network address translations and/or protocol conversions,

14 - transmitting said packets conforming to said second protocol from the said first
15 computer device to the said second computer device and

16 - decapsulating said transmitted packets conforming to said second protocol into
17 packets conforming to said first protocol.

1 2. [original] A method according to claim 1, wherein the step of taking packets
2 conforming to a first protocol and encapsulating them into packets conforming to a
3 second protocol comprises the substeps of

4 - taking packets conforming to the Internet Protocol,

5 - processing said packets according to the IPSEC protocol suite and

- 6 - encapsulating the processed packets into packets conforming to the User
7 Datagram Protocol.

- 1 3. [original] A method according to claim 1, wherein the step of taking packets
2 conforming to a first protocol and encapsulating them into packets conforming to a
3 second protocol comprises the substeps of
4 - taking packets conforming to the Internet Protocol,
5 - processing said packets according to the IPSEC protocol suite and
6 - encapsulating the processed packets into packets conforming to the
7 Transmission Control Protocol.

- B¹
1 4. [currently amended] A method according to claim 1, further comprising the
2 step of compensating for the network address translations on said second protocol in
3 the packets that are transmitted from the said first computer device to the said second
4 computer device.

- 1 5. [currently amended] A method according to claim 4, wherein said step of
2 compensating for the said network address translations comprises a step of performing
3 address translation based on the information obtained in the step of determining what
4 network address translations, if any, occur on packets transmitted between the said
5 first computer device and the said second computer device.

- 1 6. [currently amended] A method according to claim 5, wherein said step of
2 compensating for the said network address translations further comprises a step of
3 performing port number translation based on the information obtained in the step of

4 determining what network address translations, if any, occur on packets transmitted
5 between the said first computer device and the said second computer device.

1 7. [currently amended] A method according to claim 1, additionally
2 comprising the step of periodically transmitting keepalive packets between the said first
3 computer device and the said second computer device to ensure that the said network
4 address translations, if any, occurring on packets transmitted between the said first
5 computer device and the said second computer device stay the same.

B 1 8. [currently amended] A method for conditionally setting up a secure
2 communication connection between a first computer device and a second computer
3 device through a packet-switched data transmission network including comprising
4 intermediate computer devices, where at least one of said computer devices performs a
5 network address translation and/or a protocol conversion, the method comprising the
6 steps of

7 - finding out, whether or not the said second computer device supports a
8 communication method where:

9 it is determined what network address translations and/or protocol
10 conversions, if any, occur on packets transmitted between the said first
11 computer device and the said second computer device;
12 if it is found that network address translations or protocol
13 conversions occur on packets transmitted between said first computer
14 device and said second computer device, packets are taken that conform
15 to a first protocol and encapsulated into packets that conform to a second
16 protocol, which second protocol is capable of traversing network address

translations and/or protocol conversions;

said packets conforming to said second protocol are transmitted from the said first computer device to the said second computer device; and said transmitted packets conforming to said second protocol are decapsulated into packets conforming to said first protocol,

- as a response to a finding indicating that the second computer device supports said communication method, setting up a secure communication connection between the said first computer device and the said second computer device in which communication connection said communication method is employed and
- as a response to a finding indicating that the said second computer device does not support said communication method, disabling the use of said communication method between the said first and the said second computer devices.

9. [currently amended] A method for tunnelling packets between a first computer device and a second computer device through a packet-switched data transmission network comprising intermediate computer devices, where at least one of said computer devices performs a network address translation and/or a protocol conversion, the method comprising the steps of

- establishing a bidirectional tunnelling mode between the said first computer device and the said second computer device by exchanging packets conforming to a secure communication protocol,
- taking packets conforming to a first protocol and encapsulating them at the said first computer device into packets conforming to a second protocol, which second protocol is capable of traversing network address translations,
- transmitting said packets conforming to said second protocol from the said first

13 computer device to the said second computer device,
14 - decapsulating said transmitted packets conforming to said second protocol into
15 packets conforming to said first protocol at the second computer device,
16 - obtaining information about the address translations occurred on packets
17 transmitted between the said first computer device and the said second computer
18 device and
19 - using said obtained information to modify the established bidirectional tunnelling
20 mode between the said first computer device and the said second computer
21 device.

22
23 10. [original] A method according to claim 9, wherein the step of obtaining information
24 about the address translations occurred on packets transmitted between the first
25 computer device and the second computer device comprises the substeps of
26 - transmitting a packet between the first computer device and the second computer
27 device, said packet comprising a header part and a payload part, and
28 - comparing a network address transmitted in said payload part to a network address
29 transmitted in said header part in order to find out what changes have occurred on said
30 network address transmitted in said header part.

31
32 11. [original] A method according to claim 9, additionally comprising the step of
33 periodically transmitting keepalive packets between the first computer device and the
34 second computer device to ensure that the network address translations, if any,
35 occurring on packets transmitted between the first computer device and the second
36 computer device stay the same.

38 12. [original] A method according to claim 9, wherein the step of using said obtained
39 information to modify the operation of the tunnelling of packets comprises the substep of
40 introducing an address translation before the encapsulation of packets in order to
41 compensate for the network address translations that occur on packets transmitted
42 between the first computer device and the second computer device.

43
44 13. [original] A method according to claim 9, wherein the step of using said obtained
45 information to modify the operation of the tunnelling of packets comprises the substep of
46 introducing an address translation after the decapsulation of packets in order to
47 compensate for the network address translations that occur on packets transmitted
48 between the first computer device and the second computer device.

49
50 14. [original] A method for tunnelling packets between a first computer device and a
51 second computer device through a packet-switched data transmission network
52 comprising intermediate computer devices, in which data transmission network there
53 exists a security protocol comprising a key management connection that employs a
54 specific packet format for key management packets, the method comprising the steps of
55 - encapsulating data packets that are not key management packets into said specific
56 packet format for key management packets,
57 - transmitting said data packets encapsulated into the specific packet format from the first
58 computer device to the second computer device,
59 - discriminating at the second computer device the data packets encapsulated into the
60 specific packet format from actual key management packets and
61 - decapsulating the data packets encapsulated into the specific packet format.

62

63 15. [original] A method according to claim 14, wherein the step of encapsulating data
64 packets that are not key management packets comprises the substeps of
65 - encapsulating data packets that are not key management packets into a key
66 management packet format specified by the Internet Key Exchange protocol which
67 defines a certain Initiator Cookie field and
68 - inserting into the Initiator Cookie field of an encapsulated data packet a value indicating
69 that the encapsulated packet is a data packet and not a key management packet.

70
71 16. [original] A method for securely communicating packets between a first computer
72 device and a second computer device through a packet-switched data transmission
73 network comprising intermediate computer devices, where at least one of said computer
74 devices performs a network address translation and/or a protocol conversion and where
75 a security protocol exists comprising a key management connection, the method
76 comprising the steps of
77 - for determining what network address translations, if any, occur on packets transmitted
78 between the first computer device and the second computer device: establishing a key
79 management connection according to said security protocol between the first computer
80 device and the second computer device; composing an indicator packet with a header
81 part and a payload part of which both comprise the network addresses of the first
82 computer device and the second computer device as seen by the node composing said
83 packet; transmitting and receiving said indicator packet within the key management
84 connection; and comparing in the received indicator packet the addresses contained in
85 the header part and the payload part, and
86 - using the information concerning the determined occurrences of network address
87 translations to securely communicating packets between the first computer device and

88 the second computer device.

89
90 17. [original] A method according to claim 16, wherein the security protocol determines
91 a standard port number for a key management connection, and the method further
92 comprises the step of comparing in the received indicator packet a source port number
93 against said standard port number for a key management connection.

94
95 18. [original] A method for securely communicating packets between a first computer
96 device and a second computer device through a packet-switched data transmission
97 network comprising intermediate computer devices, where, at least one of said computer
98 devices performs a network address translation and/or a protocol conversion; where a
99 security protocol is acknowledged which determines transport-mode processing of
100 packets for transmission and reception; and where a high-level protocol checksum has
101 been determined for checking the integrity of received packets, the method comprising
102 the steps of
103 - at the first computer device, performing transport-mode processing for packets to be
104 transmitted to the second computer device,
105 - at the second computer device, performing transport-mode processing for packets
106 received from the first computer device, said transport-mode processing comprising the
107 decapsulation of received packets and
108 - at the second computer device, updating the high-level protocol checksum for
109 decapsulated packets for compensating for changes, if any, caused by network address
110 translations.

111
112 19. [original] A method according to claim 18, wherein

113 - the step of performing transport-mode processing at the first computer device for
114 packets transmitted to the second computer device takes the form of performing
115 transport-mode processing as determined in the IPSEC protocol suite, and
116 - the step of performing transport-mode processing at the second computer device for
117 packets received from the first computer device takes the form of performing
118 transport-mode processing as determined in the IPSEC protocol suite.
119

6¹ 120 20. [original] A method according to claim 18, additionally comprising the steps of
121 - at the first computer device, after performing transport-mode processing for a packet to
122 be transmitted to the second computer device, encapsulating the processed packet into a
123 packet conforming to a certain second protocol, which second protocol is capable of
124 traversing network address translations and
125 - at the second computer device, before performing transport-mode processing for a
126 packet received from the first computer device, decapsulating the received packet from
127 the packet conforming to said second protocol and replacing a number of network
128 addresses in the decapsulated packet with a corresponding number of network
129 addresses taken from the received packet before decapsulation.
130

131 21. [original] A method according to claim 18, wherein the step of updating the
132 high-level protocol checksum takes the form of recomputing the checksum for the
133 transport-mode-processed packets.
134

135 22. [original] A method according to claim 18, wherein the method additionally
136 comprises the step of obtaining information about the network addresses of the first and
137 second computer devices before and after network address translations, and the step of